University of Bahrain
College of Information Technology
Department of Computer Science
Semester 1, 2012-2013
ITCS216 (Data Structures and Algorithms)

Test1 Exam

Date: November 5, 2012 **Time**: 11:00-12:30

STUDENT NAME	.....
STUDENT ID #	.....

NOTES:

- WRITE ONLY ONE SOLUTION FOR EACH QUESTION
- SWITCH OFF YOUR MOBILE PHONES
- THIS EXAM CONTAINS 6 PAGES

QUESTION #	MARKS		COMMENTS
1	6		
2	10		
3	14		
TOTAL	30		

Question 1 (3-3 marks)

Prove the following statements:

1.1. 2^{n+a} is $O(2^n)$

$2^{n+a} = 2^a.2^n \leq c2^n$ where $c \geq 2^a$

So, $\forall c \geq 2^a$ and $n \geq 0$ $2^{n+a} \leq c2^n$

Therefore, 2^{n+a} is $O(2^n)$

1.2. $\sum_{i=1}^{n} 2i$ is $\Theta(n^2)$

$\sum_{i=1}^{n} 2i = 2\sum_{i=1}^{n} i = \frac{2n(n+1)}{2} = n(n+1)$

$n^2 \leq n(n+1) \leq 2n^2 \quad \forall n \geq 1$

So, $\sum_{i=1}^{n} 2i$ is $O(n^2)$ and $\Omega(n^2)$

Therefore, $\sum_{i=1}^{n} 2i$ is $\Theta(n^2)$

$j = n-1 \rightarrow j = n-2 \rightarrow j = n-3, \ldots \rightarrow j = i+1$

$= n-1-(i+1)+1$

$= n-1-i-1+1$

$= n-1-i$

Question 2 (3-3-4 marks)

Given the following java function:

```
1. public void bubbleSort(int []data)
2. {
3.    for(int i = 0; i < data.length-1; i++)
4.       for(int j = data.length-1; j > i; j--)
5.          if(data[j] < data[j-1])
6.             swap(data, j, j-1);
7. }
```

2.1. Find the total number of comparisons performed in the statement number 5. Suppose that the length of the array data is n.

The outer for loop runs $n-1$ times

The inner for loop runs $n-i-1$ times

So, the number of comparisons in statement 6 is $\sum_{i=0}^{n-2}(n-i-1) = \sum_{i=1}^{n-1} i = \frac{n(n-1)}{2}$

2.2. Find the total number of assignments performed in the line 6 in the best case and in the worst case. Suppose that swap is a function that accepts as parameters an array and the positions of two elements to be swapped.

Best case: if the array is sorted in increasing order, 0 swap

Worst case: if the array is sorted in decreasing order, there are

$3\sum_{i=0}^{n-2}(n-i-1) = \frac{3n(n-1)}{2}$ swaps

1.1. Find the asymptotic complexity in the worst case of the function bubbleSort.

`i = 0;` → 1

`i < data.length-1;` → n

`i++ and j = data.length-1;` → $(n-1)+(n-1)$

`j > i;` → $1+\frac{n(n-1)}{2}$

`j--` → $\frac{n(n-1)}{2}$

`if(data[j] < data[j-1])` → $\frac{n(n-1)}{2}$

`swap(data, j, j-1);` → $\frac{3n(n-1)}{2}$

Total number of iterations: $2(n-1)(2+3n) = O(n^2)$

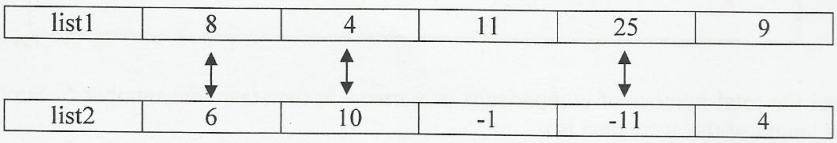
Question 3 (14 marks)

Write a function called **swapListX** that accepts as parameters:

- Two objects list1 and list2 of type ArrayList,
- An integer number X.

Both lists list1 and list2 have the same size. The function compares the elements of the same index and swap them if their sum is equal to X. The function should return the number of swaps.

Example: X = 14




```
public static int swapListX(ArrayList<Integer> list1, ArrayList<Integer> list2, Integer X)
  {
    int pos, i, n = 0;;
    Integer element1, element2;
    for(i = 0; i<list1.size(); i++)
    {
      element1 = list1.get(i);
      element2 = list2.get(i);
      if(element1 + element2 == X)
      {
            n++;
            list1.set(i, element2);
            list2.set(i, element1);
      }
    }
    return n;
  }
```